

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS/
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problem Mailbox.**

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 02-210542
(43)Date of publication of application : 21.08.1990

(51)Int.Cl.

G06F 9/46
G06F 15/16

(21)Application number : 01-031564
(22)Date of filing : 10.02.1989

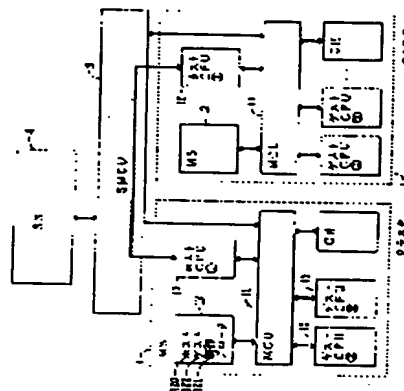
(71)Applicant : FUJITSU LTD
(72)Inventor : KUMANO TAKESHI
UEDA KOICHI

(54) EXECUTION CONTROL SYSTEM FOR VIRTUAL COMPUTER SYSTEM

(57)Abstract:

PURPOSE: To carry out the host and guest processes independently of each other in order to eliminate the overhead and to attain the flexible control by providing a means which applies an interruption to a host mechanism from a virtual (guest) computer when the communication is required to the host mechanism.

CONSTITUTION: A means 11 which dynamically assigns a host mechanism 1 which instructs the start of execution of a virtual computer 2 to a specific CPU 10 in each cluster 1, a means 11 applies an interruption to the host mechanism 1 from the computer 2 when the mechanism 1 assigns selectively the execution of the computer 2 to its own CPU 10 or another CPU 10 and the working computer 2 needs the communication to the mechanism 1 are provided. Then the host and guest processes are carried out independently of each other via plural CPUs 10. In such a constitution, the host overhead caused by the inter-host interference is reduced in a virtual computer system. Then the more flexible control of resources is attained and the load can be decentralized.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

⑩ 日本国特許庁(JP)

⑪ 特許出願公開

⑫ 公開特許公報(A) 平2-210542

⑬ Int. Cl.³

G 06 F 9/46
15/16

識別記号

3 5 0
4 3 0

庁内整理番号

8945-5B
6745-5B

⑭ 公開 平成2年(1990)8月21日

審査請求 未請求 請求項の数 3 (全10頁)

⑮ 発明の名称 仮想計算機システムにおける実行制御方式

⑯ 特 願 平1-31564

⑰ 出 願 平1(1989)2月10日

⑱ 発 明 者 熊 野 剛 神奈川県川崎市中原区上小田中1015番地 富士通株式会社内

⑲ 発 明 者 上 田 孝 一 神奈川県川崎市中原区上小田中1015番地 富士通株式会社内

⑳ 出 願 人 富士通株式会社 神奈川県川崎市中原区上小田中1015番地

\r\n㉑ 代 理 人 弁理士 井 桁 貞 一

明 細 書

1. 発明の名称

仮想計算機システムにおける実行
制御方式

2. 特許請求の範囲

(1) 複数の物理中央処理装置(CPU)(10)から
構成されたクラスタ(1)を複数個備えた仮想計算
機システムにおいて、

該仮想計算機システム上で走行する仮想計算機
(ゲスト)(2)の実行開始を指示するホスト機構
(11)を、上記各クラスタ(1)内の特定の中央処
理装置(CPU)(10)に動的に割当る手段と、

該割当てられた中央処理装置(CPU)(10)上で走
行するホスト機構(11)は、上記仮想計算機(ゲ
スト)(2)の実行を、自己の中央処理装置(CPU)
(10)、又は、他の中央処理装置(CPU)(10)に選択
的に割当る手段と、

上記割当てられた特定の中央処理装置(CPU)(1
0)で走行している仮想計算機(ゲスト)(2)に

おいて、上記ホスト機構(11)に対して通信の必
要が生じた場合には、該仮想計算機(ゲスト)(
2)から上記ホスト機構(11)に割込む手段とを
備えて、

ホスト処理とゲスト処理とを、複数の中央処
理装置(CPU)(10)で独立に実行することを特徴と
する仮想計算機システムにおける実行制御方式。

(2) 上記計算機システムにおいて、仮想計算機
(ゲスト)(2)間の通信をクラスタ(1)間で行う
場合には、上記ホスト機構(11)の割当てられて
いる中央処理装置(CPU)(10)を介して行うことを
特徴とする請求項1に記載の仮想計算機システム
における実行制御方式。

(3) 上記ホスト機構(11)、又は、ゲスト(2)
を選択的に特定の中央処理装置(CPU)(10)に割当
る手段として、少なくとも、ゲスト識別子、又は、
割込み識別子を設定するフィールドを備えた特定
の命令を設けることを特徴とする請求項1に記載
の仮想計算機システムにおける実行制御方式。

3. 発明の詳細な説明

〔目次〕

概要

産業上の利用分野

従来の技術と発明が解決しようとする課題
課題を解決するための手段

作用

実施例

発明の効果

〔概要〕

複数の物理中央処理装置(CPU)から構成されたクラスタを複数の備えた仮想計算機システムにおける仮想計算機(ゲスト)の実行制御方式に関し、

仮想計算機システムにおけるホスト間の干渉によるホストでのオーバーヘッドを低減させ、又、より柔軟な資源管理、及び、負荷分散を図ることを目的とし、

(1)該仮想計算機システム上で走行する仮想計算

機(ゲスト)②の実行開始を指示するホスト機構①を、上記各クラスタ内の特定の中央処理装置(CPU)に動的に割当る手段と、該割当てられた中央処理装置(CPU)上で走行するホスト機構①は、上記仮想計算機(ゲスト)②の実行を、自己の中央処理装置(CPU)、又は、他の中央処理装置(CPU)に選択的に割当る手段と、上記割当てられた特定の中央処理装置(CPU)で走行している仮想計算機(ゲスト)②において、上記ホスト機構①に対して通信の必要が生じた場合には、該仮想計算機(ゲスト)②から上記ホスト機構①に割込む手段とを備えて、ホスト処理とゲスト処理とを、複数の中央処理装置(CPU)で独立に実行するように構成する。(2)上記計算機システムにおいて、仮想計算機(ゲスト)②間の通信をクラスタ間で行う場合には、上記ホスト機構①の割当てられている中央処理装置(CPU)を介して行うように構成する。(3)上記ホスト機構①、又は、ゲスト②を選択的に特定の中央処理装置(CPU)に割当る手段として、少なくとも、ゲスト識別子、又は、割込み識別子

- 3 -

を設定するフィールドを備えた特定の命令を設けるように構成する。

〔産業上の利用分野〕

本発明は、複数の物理中央処理装置(CPU)から構成されたクラスタを複数の備えた仮想計算機システムにおける仮想計算機(ゲスト)の実行制御方式に関する。

最近のデータ処理の多様化、複雑化に伴って、計算機システムに対する処理能力の向上要求には際限がないが、単体の中央処理装置(CPU)では処理能力に限界があることから、最近では、複数の中央処理装置(CPU)を備えたマルチプロセッサシステムが構築されるようになってきている。

一方、従来から、資源の有効利用の面から1個の物理中央処理装置(CPU)上で複数のゲストプログラム(仮想計算機)を実行させる仮想計算機システムが知られているが、上記計算機システムのマルチプロセッサ化に伴い、該マルチプロセッサシステム上で複数の仮想計算機を走行させる

仮想計算機システムが考えられるようになってきている。

このような、複数の中央処理装置(CPU)からなるマルチプロセッサシステム上においては、複数の仮想計算機(ゲスト)を、ホストの制御の基に、該ホストでのオーバーヘッドを増加させることなく、且つ、柔軟に、共有資源の利用、負荷分散を図ることができるように行う制御方式が必要とされる。

〔従来の技術と発明が解決しようとする課題〕

第3図は従来の仮想計算機システムにおける実行制御方式を説明する図であり、(a)はシステム構成の例を示し、(b1)は実行制御方式を模式的に示し、(b2)は動作タイムチャートを示している。

一般に、マルチプロセッサシステムにおいては、データ処理の高速化を実現する為に、並列処理手段が用いられるが、ホストプログラム(以下、ホストという)①においても、同じ思想を踏襲して、該マルチプロセッサシステム上で仮想計算機シス

- 5 -

- 6 -

テムを構築する場合、該複数個の各中央処理装置（以下、CPU という）10に、該ホスト①を割当てており、ゲストプログラム（以下、ゲストという）②は、その実行開始を指示したホスト①が走行している物理 CPU上で走行していた。

上記の各ゲスト②の実行要求は、例えば、入出力装置(I/O) 13からのデータ転送要求割込み、データ転送終結報告割込み等の入出力割込み、CPU 10からの割込み、或いは、図示されていないキーボード(KB)からの人手割込み等で発生する。

該入出力割込みにおいては、ホスト①が発行する入出力命令に指定されているゲスト識別子を、該入出力処理終了時の、該入出力割込み要求信号に付加して該入出力割込み要求を発生するように構成されており、該発生した入出力割込み要求は、チャネル装置(CH) 13を介して、例えば、記憶制御装置(MCU) 11内に設けられている割込みハードウェア(IHW) 11aに入力される。

該割込みハードウェア(IHW) 11aにおいては、上記ゲスト識別子を基に、主記憶装置(MS) 12上

の対応ゲスト②の制御ブロック（ゲストA、B～）121を参照して、そのマスク情報を読み取り、今発生した割込み要求の割込み可否を判定し、割込み可能な状態であると、該ゲスト（ゲストA、B～）②が走行しているCPU 機番をゲスト実行テーブル 11bを検索して認識し、対応CPU 10に割込みを行う。

このとき、該ゲスト②が走行していない時には該割込み要因は、上記割込みハードウェア(IHW) 11a内に保持される。

各CPU(10～) 10では現在実行中のゲスト②の実行が割込み等で中断、又は、終了すると、制御がホスト①に移り、ホスト①は、該中断要因が割込みの場合には、その割込み要因に対応したゲスト②を、その他の場合には、上記記憶制御装置(MCU) 11内の該ゲスト対応で保留されている入出力割込み要求、ゲスト要求等を検索し、その1つを取り出してゲスト識別子を認識し、対応するゲスト②の制御ブロック 121を取り出して自CPU 10内の汎用レジスタ、制御レジスタ等に設定し、特定

- 7 -

のゲストスタート(SGUST) 命令を発行して、該ゲスト②を走行させると共に、上記ゲスト実行テーブル 11bの該ゲスト識別子に対応した該当領域に自己の物理 CPU機番を設定することで、該ゲスト②に対する割込み要求、或いは、ゲスト実行要求を実行させることができる。

このような従来方式の仮想計算機システムでの実行制御方式を、特定のCPU A に注目して模式的に示すと、本図 (b1)、(b2)に示すようになる。

即ち、CPU A上で走行しているホスト①は、上記のようにして、これから実行すべきゲスト②を捕捉すると、スタートゲスト命令(SGUEST)を発行し、該命令中のオペランドが指示するゲスト識別子に基づいて、主記憶装置(MS) 12から対応する制御ブロック 121を読み取り、自CPU A内のプログラムステータス語(PSW)、汎用レジスタ、制御レジスタ等に設定して、該ゲスト②を走行させ、該ゲスト②の実行が終了すると、例えば、ゲスト②自身の動作に起因する割込みの発生等によって、元のホスト①に戻るよう動作する。(b2)の動

作タイムチャート参照)

上記において説明した、マルチプロセッサシステムで、各CPU 10に割振られてホスト①によって、複数個のゲスト②を実行する方式は、あくまでも1例であって、これに限定されるものでないことはいふまでもないことである。

例えば、上記記憶制御装置(MCU) 11内にゲスト割当てカウンタを設けて、ゲスト実行要求を順番に均等に割当てる方法も考えられる。

この場合には、あるゲスト②に対する割込みが発生しても、該ゲストが走行している物理 CPUに割込みができなくなる問題がある。

又、各ホスト①が、それぞれ、個別に、記憶制御装置(MCU) 11に対して、実行待ちのゲスト②を要求する場合もある。この場合には、ホスト①間で競合が起こるので、例えば、優先順位により、優先度の高いホスト①がゲスト②を確保する。

その他、色々変形が考えられるが、何れにしても、複数個のCPU 10に、ホスト①が分散している従来方式においては、該仮想計算機システム中で

- 9 -

- 10 -

共有され、排他制御を必要とする資源、例えば、主記憶装置(MS) 12 上の共有データ(例えば、上記の制御ブロック 121) に対するアクセス等で、ホスト①同士で待ち合わせが必要となる等のオーバーヘッドが多くなるという問題があった。更に、上記のように、ホスト①と、ゲスト②とが同一のCPU 10上で走行している為、CPU 10間での負荷の分散等の制御が困難になるという問題があった。

本発明は上記従来の欠点に鑑み、複数個の物理中央処理装置(CPU) から構成されたクラスタを複数個備えた仮想計算機システムにおいて、仮想計算機(ゲスト)の実行制御を行うのに、該仮想計算機システムにおけるホスト間の干渉によるホストでのオーバーヘッドを低減させ、又、より柔軟な資源管理、及び、負荷分散を図ることができる仮想計算機システムにおける実行制御方式を提供することを目的とするものである。

(課題を解決するための手段)

第1図は本発明の原理説明図であり、(a) はシ

ステム構成例を示し、(b1)は実行制御方式を模式的に示し、(b2)は動作タイムチャートを示し、(b3)は命令フォーマットの一例を示している。

上記の問題点は下記の如くに構成された仮想計算機システムにおける実行制御方式によって解決される。

(1) 複数個の物理中央処理装置(CPU) 10から構成されたクラスタ 1を複数個備えた仮想計算機システムにおいて、

該仮想計算機システム上で走行する仮想計算機(ゲスト) ②の実行開始を指示するホスト機構①を、上記各クラスタ 1内の特定の中央処理装置(CPU) 10に動的に割当る手段と、

該割当てられた中央処理装置(CPU) 10上で走行するホスト機構①は、上記仮想計算機(ゲスト) ②の実行を、自己の中央処理装置(CPU) 10、又は、他の中央処理装置(CPU) 10に選択的に割当る手段と、

上記割当てられた特定の中央処理装置(CPU) 10で走行している仮想計算機(ゲスト) ②において、

- 1 1 -

上記ホスト機構①に対して通信の必要が生じた場合には、該仮想計算機(ゲスト) ②から上記ホスト機構①に割込む手段とを備えて、

ホスト処理とゲスト処理とを、複数個の中央処理装置(CPU) 10で独立に実行するように構成する。

(2) 上記計算機システムにおいて、仮想計算機(ゲスト) ②間の通信をクラスタ 1間で行う場合には、上記ホスト機構①の割当てられている中央処理装置(CPU) 10を介して行うように構成する。

(3) 上記ホスト機構①、又は、ゲスト②を選択的に特定の中央処理装置(CPU) 10に割当る手段として、少なくとも、ゲスト識別子、又は、割込み識別子を設定するフィールドを備えた特定の命令を設けるように構成する。

(作用)

即ち、本発明によれば、複数個の物理中央処理装置(CPU) から構成されたクラスタを複数個備えた仮想計算機システムにおいて、ホスト①を該複数個の物理CPU 内の特定のCPU に割り当てるよう

に機能させる。

この割当ての際、例えば、電源投入時等においては、固定的に、特定のCPU にホスト①を割当てるが、稼働中においては、該ホスト①が割当てられているCPU において、例えば、ホストの仕事が無くなった時、記憶制御装置(MCU) 内に設けられている、例えば、ゲスト要求キューを参照して、特定のゲスト②を選択し、記憶制御装置(MCU) 内に設けられているホスト実行テーブルをクリアすると共に、CPU ビジーテーブル中の自己のCPU 機番欄に該実行を始めたゲスト②のゲスト識別子を設定した後、該物理 CPU において該ゲスト②を実行するようにしてもよい。

すると、他のゲスト②を実行していた物理CPU がホストに制御を渡したくなったとき、上記ホスト実行テーブルを参照して、ホスト①を実行している物理CPU が存在しないことを認識した時、自己がホスト①を実行し、上記ホスト実行テーブルに自己のCPU 機番を登録するようにすることで、該ホスト①を複数個のCPU に動的に割当ることが

- 1 2 -

- 1 3 -

- 1 4 -

できるようになる。

又、ホスト①は、特定のゲスト②の実行を開始させる際、該ホスト①が発行する本発明の特定のスタートゲスト命令(SGUEST)が指示するゲスト識別子に基づいて、該ゲスト②の状態を特定する各種制御情報等を含む制御ブロックを主記憶装置(HS)から読み取り、該制御ブロックと共に、該ゲスト②を走行させる物理CPUを、例えば、上記記憶制御装置(MCU)内に設けられているCPU ビジーテーブルを参照して、空きの物理CPUに該ゲスト②の走行を指示する。

この時、該ゲスト②が走行する物理CPUはホスト①が走行しているCPUであってもよい。

該ホスト①は上記走行指示と共に、上記CPU ビジーテーブル中の該当CPU 機番の欄に、該実行指示したゲスト②のゲスト識別子を設定する。

このようにして、ホスト①から、ゲスト②を走行させる物理CPUを自己のCPUと異なる物理CPUに指示した場合には、該ホスト①は該ゲスト②の実行が、該指示した物理CPUで実際に開始される

ことを待つことなく、次の命令の実行に移ることができる。

こうして、該ゲスト②が上記特定のスタートゲスト命令(SGUEST)で指示された物理CPUで走行しているとき、ホスト①に対して何らかの通信の必要性が生じた場合には、該ゲスト②からホスト①に割込みを行うことで該通信を実行する。

このとき、該ゲスト②を実行していた物理CPUは、割込みハードウェア(IHW)に割込み要求を依頼する。該割込みハードウェア(IHW)は、上記ホスト実行テーブルを見て、該ホスト①が走行している物理CPUの機番を認識し、該物理CPUに割込みを行う。

又、あるゲスト②から他のゲスト②に割込みを行う場合でも、同様にして、上記CPU ビジーテーブル中に記憶されている現在走行中のゲスト識別子を認識して、対応するゲスト識別子を持つ物理CPUに割込みを行うことで、ホスト①に介入することなく、直接実行形式の割込みを行うことができる。若し、該CPU ビジーテーブル中に、割込み

- 15 -

先のゲスト識別子を検索できない場合には、該割込み要求は保留されることになる。上記ゲスト間通信は、例えば、物理CPU機番と、ゲスト識別子とを指示する公知のシグナルプロセッサ命令を用いてもよい。

又、クラスタ間通信を行う場合には、上記ホスト①の走行している物理CPUが、例えば、上記公知のシグナルプロセッサ命令形式のシグナルクラスタ命令を発行することで、上記クラスタ内でのゲスト間通信と同様の処理手順で、該クラスタ間通信を行うことができる。

このように機能するので、ホスト間の干渉によるオーバーヘッドが解消すると共に、システム中でホスト①が高々一つしか存在しないので、共有資源の管理が容易となり、又、実行制御が単純になる。又、ゲスト②が走行する物理CPUを、ホスト①と共有することなく、専用化できる為、キャッシュメモリを備えた物理CPUにおいては、ヒット率を向上させることができる効果がある。

- 17 -

- 16 -

〔実施例〕

以下本発明の実施例を図面によって詳述する。

前述の第1図が本発明の原理説明図であり、第2図は本発明の一実施例を模式的に示した図であって、特定の物理CPU 10に割当てられたホスト①において、特定のスタートゲスト命令(SGUEST)を発行し、そのオペランドで特定のゲスト②を指定して、該ゲスト②を、例えば、空いている物理CPU 10で、或いは、自己の物理CPU 10、或いは、強制的に解放させた物理CPU 10で走行させ、該走行中のゲスト②において、ホスト①に対して何らかの通信の必要性が生じた場合、該ゲスト②よりホスト①が走行している物理CPU 10に割込む手段が本発明を実施するのに必要な手段である。尚、全図を通して同じ符号は同じ対象物を示している。

以下、第1図を参照しながら、第2図によって本発明の仮想計算機における実行制御方式を説明する。

まず、本実施例においては、例えば、専用のス

- 18 -

スタートゲスト(SGUEST)命令を設ける。

前述のように、電源投入時等の初期化処理において、予め、定められている物理 CPU 10 がホスト①に指定され、ホスト実行テーブル 110中に登録される。

該指定された物理 CPU (以下、ホスト CPU という) 10 は、例えば、記憶制御装置(MCU)11 内に設けられているゲスト要求キュー 111を検索して、既に、実行要求を出しているゲスト②を捕捉すると、本発明のスタートゲスト(SGUEST)命令を発行する。

該スタートゲスト(SGUEST)命令が、該ホスト CPU 10 で発行されると、該命令のオペランドが指示するゲスト識別子(第1図(b3)参照)を基に、主記憶装置(MS)12上の、予め、登録されている該ゲスト識別子対応に設けられている制御ブロック 121を読み取ると共に、空いている物理 CPU 10を検索して、該当の物理 CPU 10に当該ゲスト②の実行を指示する。(第1図(b1),(b2)参照)

該ホスト CPU 10からの指示に基づいて、指示

されたゲスト②を実行する物理 CPU 10を、ゲスト CPU という。該ゲスト CPU 10は、上記制御ブロック 121の情報を自己の物理 CPU 10内に設定して、該ゲスト②を実行する。

このようにして、ホスト CPU 10と該ゲスト CPU 10とが異なる場合には、ホスト①は該指示したゲスト②が指定した上記ゲスト CPU 10において実際に実行が開始されるのを待つことなく、次の処理に移ることができる。

上記の動作において、該空きのゲスト CPU 10が無ければ、例えば、自己の物理 CPU、即ち、上記、ホスト CPU 10において、該ゲスト②に制御を移すようにしてもよいし、適当なゲスト②の実行を途中で中止させて、その物理 CPU 10を強制的に解放し、該ゲスト②を実行させてもよい。

上記ホスト CPU 10でゲスト②を実行したときには、上記ホスト実行テーブル 110はクリアされる。

このような状態にあるときに、他のゲスト CPU 10で、あるゲスト②が走行しているゲスト CPU

- 19 -

10からホスト①への通信が必要になったとき、該ゲスト②は、ホスト①への割込みを要求すると、該割込み要求は、例えば、割込みハードウェア(IHW)11aに入力される。

該割込みハードウェア(IHW)11aにおいては、上記ホスト実行テーブル 110を参照して、現在、ホスト①が走行しているホスト CPU 10の機番がないことを知り、該要求元のゲスト CPU 10にその旨を通知する。該通知を受けたゲスト CPU 10はホスト①を実行して、該ゲスト②からの割込みを処理することで、ホスト①の物理 CPUへの動的な割当てが実現される。

又、この時、ホスト①が特定の物理 CPU 10で走行中の時には、該ホスト CPU 10に割込みを行うことで、ゲスト②からホスト①への通信を行うことができる。

又、図示されていないキーボード等からのゲスト要求が生起すると、図示されていないハードウェアにおいて、要求要因が解析され、ゲスト要求と認識されることで、ゲスト要求キュー 111にゲ

スト識別子と共に登録される。

該ゲスト要求キュー 111に登録されているゲスト要求は、前述のように、一般には、ホスト CPU 10によって検索され、本発明のスタートゲスト(SGUEST)命令が発行されることで、特定の物理 CPU 10に実行が指示される。

上記の実施例において、本発明のスタートゲスト(SGUEST)命令が発行されたときの動作は、例えば、記憶制御装置(MCU)11内のハードウェア機構で、該命令のオペランドが指示するゲスト識別子に基づいて、該ゲスト識別子に対応したゲスト②を走行させる物理 CPU 10を検索する方法で説明したが、例えば、ホスト①が、予め、該ゲスト②を走行させるべき物理 CPU 10を認識した後、該スタートゲスト(SGUEST)命令に、物理 CPU機番と、ゲスト識別子とを指示する(公知のシグナルプロセッサ命令形式)ことで、対応する物理 CPU 10に直接的に、該ゲスト②を走行させるようにしてもよい。

又、割込み方法についても、同様にして、該シ

- 22 -

- 21 -

グナルプロセッサ命令と同じフォーマット形式の命令を生成して、直接的に、該当の物理 CPU 10 に割込ませることもできる。

上記の実施例は、第 2 図に示したクラスタ 1 内での動作として説明しているが、あるクラスタ 1 から他のクラスタ 1 に通信する場合には、ホスト CPU 10 がゲスト要求を取り出し、該ゲスト要求が他のクラスタ 1 への要求であることを認識した時には、例えば、上記シグナルプロセッサ命令と同じフォーマット形式の、例えば、シグナルクラスタ命令を設けて、該命令で相手のクラスタ番号、ゲスト識別子、或いは割込み要因を指示して、上記記憶制御装置(MCU) 11 から、クラスタ 1 間のシステム記憶制御装置(SMCU) 3 を介して、上記クラスタ 1 内での動作と同じような処理手順で、該他のクラスタ 1 の記憶制御装置(MCU) 11 に送出する。(第 1 図 (a) 参照)

該シグナルクラスタ命令を受信したクラスタ 1 の記憶制御装置(MCU) 11 においては、該クラスタ 1 内のホスト CPU 10 に該命令を実行させ、該ク

ラスタ 1 内での処理に必要なスタートゲスト(SGU EST)命令を生成・発行することで、クラスタ間通信、或いは、クラスタ間割込みを、それぞれのホスト CPU 10 を介して行うことができる。

このように、本発明は、複数の物理中央処理装置(CPU) から構成されたクラスタを複数個備えた仮想計算機システムにおいて、ホスト①、ゲスト②を実行制御するのに、特定の物理 CPU に該ホスト①を割当て、該ホスト①が割当てられている物理 CPU、又は、他の物理 CPU にゲスト②を割当るようにして、該ホスト①とゲスト②とを独立の物理 CPU で実行させるようにし、該仮想計算機システム中に高々 1 つのホスト①しか存在しないようにした所に特徴がある。

(発明の効果)

以上、詳細に説明したように、本発明の仮想計算機システムにおける実行制御方式は、複数の物理中央処理装置(CPU) から構成されたクラスタを複数個備えた仮想計算機システムにおいて、(1)

- 2 3 -

該仮想計算機システム上で走行する仮想計算機(ゲスト)②の実行開始を指示するホスト機構①を、上記各クラスタ内の特定の中央処理装置(CPU) に動的に割当る手段と、該割当てられた中央処理装置(CPU) 上で走行するホスト機構①は、上記仮想計算機(ゲスト)②の実行を、自己の中央処理装置(CPU)、又は、他の中央処理装置(CPU) に選択的に割当る手段と、上記割当てられた特定の中央処理装置(CPU) で走行している仮想計算機(ゲスト)②において、上記ホスト機構①に対して通信の必要が生じた場合には、該仮想計算機(ゲスト)②から上記ホスト機構①に割込む手段とを備えて、ホスト処理とゲスト処理とを、複数の中央処理装置(CPU) で独立に実行するように構成する。(2)上記計算機システムにおいて、仮想計算機(ゲスト)②間の通信をクラスタ間で行う場合には、上記ホスト機構①の割当てられている中央処理装置(CPU) を介して行うように構成する。(3)上記ホスト①、ゲスト②の指定を行うのに、特定の命令を設けるようにしたものである、ホ

スト間の干渉によるオーバーヘッドが解消すると共に、システム中でホスト①が高々一つしか存在しないので、共有資源の管理が容易となり、又、実行制御が単純になる。又、ゲスト②が走行する物理 CPU を、ホスト①と共有することなく、専用化できる為、キャッシュメモリを備えた物理 CPU においては、ヒット率を向上させることができる効果がある。

4. 図面の簡単な説明

第 1 図は本発明の原理説明図。

第 2 図は本発明の一実施例を模式的に示した図。

第 3 図は従来の仮想計算機システムにおける実行制御方式を説明する図。

である。

図面において、

1 はクラスタ、

10 は中央処理装置(CPU)、又は、ホスト CPU、ゲスト CPU、又は、物理 CPU、

- 2 6 -

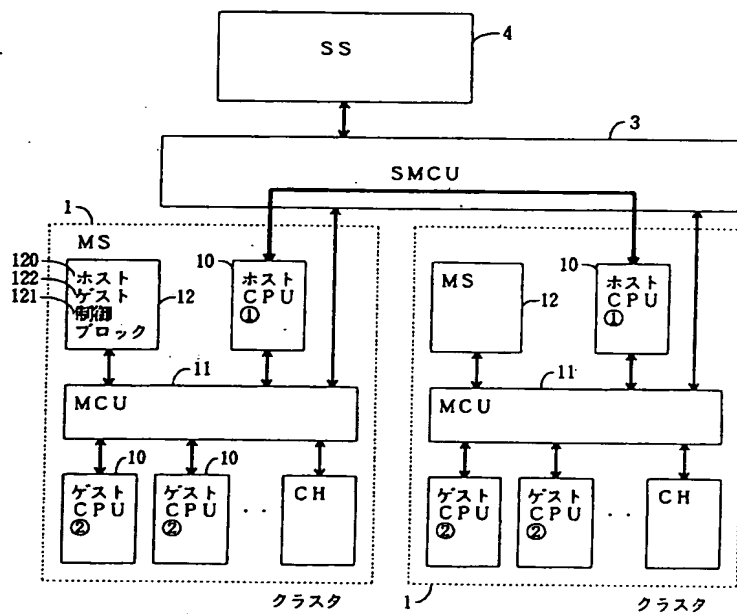
- 2 5 -

11は記憶制御装置(MCU)、12は主記憶装置(MS)、
 11aは割込みハードウェア(IRW)、
 11bはゲスト実行テーブル、
 110はホスト実行テーブル、
 111はゲスト要求キュー、
 112はCPUビジーテーブル、
 120、①はホストプログラム(ホスト)、
 121は制御ブロック、
 122、②はゲストプログラム(ゲスト)
 3はシステム記憶制御装置(SMCU)、
 4はシステム記憶装置(SS)、
 をそれぞれ示す。

代理人 弁理士 井桁貞一



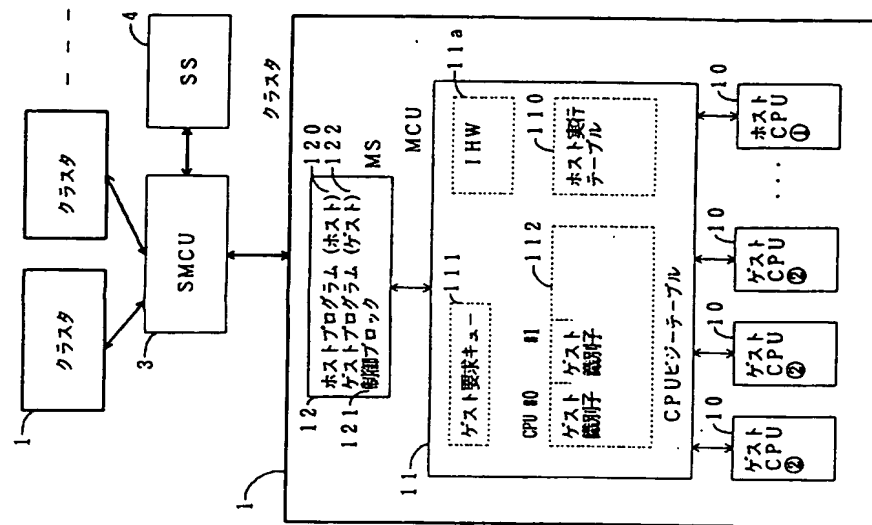
- 27 -



(a)

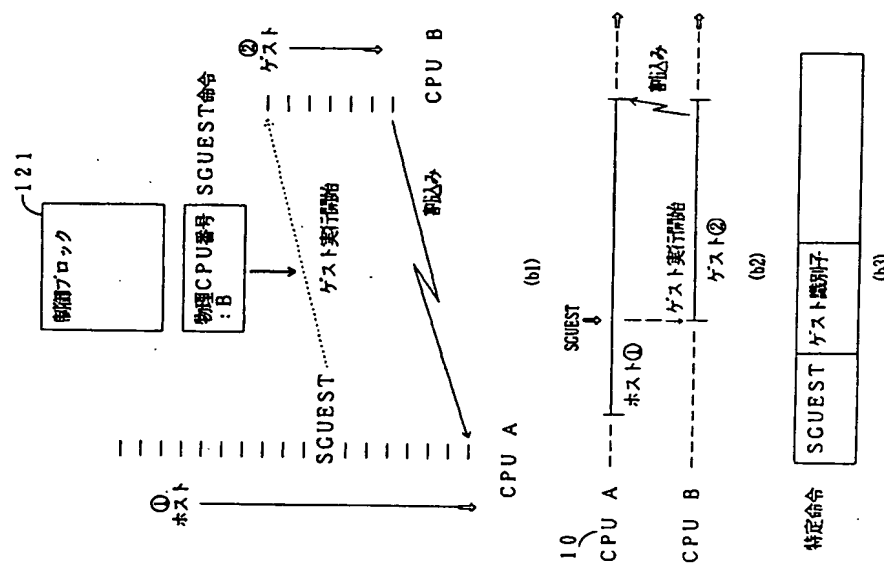
本発明の原理説明図

第1図(その1)



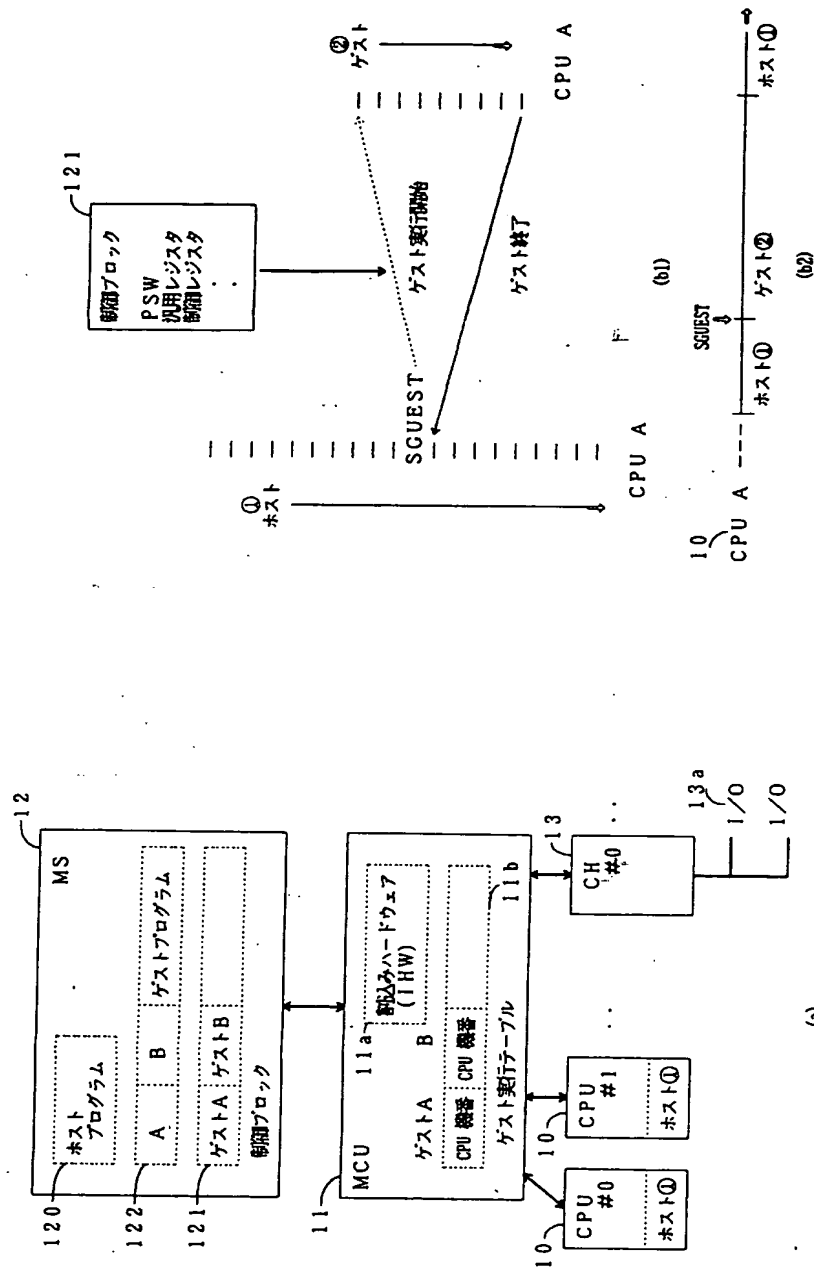
本発明の一実施例を模式的に示した図

第 2 図



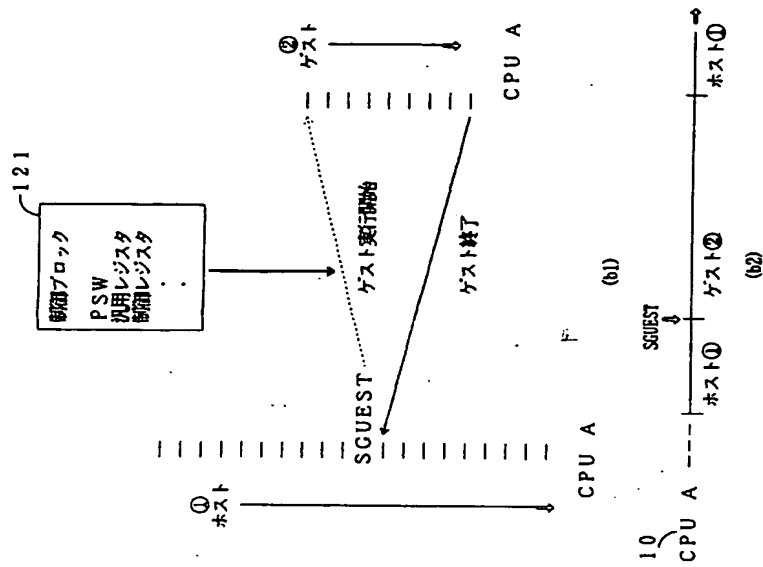
本発明の原理説明図

第1図(その2)



(a)
従来の仮想計算機システムにおける実行制御方式を説明する図

第 3 図 (その 1)



従来の仮想計算機システムにおける実行制御方式を説明する図

第 3 図 (その 2)